UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/827,033	04/05/2001	Humberto A. Sanchez II	10006819-1	4844

7590 06/18/2004

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, CO 80527-2400

EXAMINER	
LAO, SUE X	
ART UNIT	PAPER NUMBER
2126	3

DATE MAILED: 06/18/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/827,033	**Applicant(s)** SANCHEZ, HUMBERTO A.	
	Examiner S. Lao	**Art Unit** 2126	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)□ Responsive to communication(s) filed on _____.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-20 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-6, 8-20 is/are rejected.

7)☒ Claim(s) 7 is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 2.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. Claims 1-20 are presented for examination.

2. The disclosure is objected to under 37 CFR 1.71, as containing terminology which is so different from that which is generally accepted in the art to which this invention pertains. For example, throughout the application, applicant describes C++ as a procedural language. However, C++ is conventionally recognized as an object-oriented language because of its encapsulation, inheritance and polymorphism.

Applicant is required to provide a clarification of these matters or correlation with art-accepted terminology so that a proper comparison with the prior art can be made. Applicant should be careful not to introduce any new matter into the disclosure (i.e., matter which is not supported by the disclosure as originally filed).

3. Claims 4, 13 and 19 are objected to because of the following informalities: C++ is not typically characterized as a procedural language, as discussed in section 2 of this action.

4. For the reasons discussed in sections 2 and 3 of this action, the following art rejections assume the claimed procedural programming language / C++ referring to legacy object-oriented programming language / C++, as best understood and as it appears to be.

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 1-5, 8-14, 16-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nevarez et al (U S Pat. 6,609,158) in view of Loon ("Callback functions: The heart of every GUI's API").

As to claim 16, Nevarez teaches a computer system that enables the mapping of procedural code to object oriented classes, comprising:

a memory (col. 9, lines 6-22);

a processor (col. 8, lines 39-51) that runs an application (universal component system (UCS) product), wherein the application includes:

a program in a procedural programming language process space (C/C++ program 206); and,

a mapping layer (UCS 224) in an object-oriented programming language process space (UCS 224 includes object model adapter 230), wherein the mapping layer comprises entry-points (language templates) that have corresponding algorithms that invoke object-oriented class instantiation methods (callable object instantiation function) and/or remote method invocations ("RMIs") (provider 238 for Java/RMI) (col. 4, lines 16-33; col. 9, lines 25-49; col. 11, lines 36 – col. 12, line 5). It is noted that because Nevarez provides the GUI and the mapping layer, generating the GUI and the mapping layer would have been inherent, or obvious.

Nevarez does not explicitly teach the program in C/C++ includes a graphical user interface ("GUI"), wherein users enter commands through the GUI. This, however, would have been an obvious choice because the client devices of Nevarez include types of PC, laptop, workstation, dumb terminal and PDA, each of which typically interacts with a user via a GUI, and because that a primary use of C++ has been providing GUIs for such devices/systems. This is also taught by Loon (page 43, right col., 2nd - 3rd para.s). Therefore, it would have been obvious to include a GUI for a user to enter command(s) to the program. One of ordinary skill in the art would have been motivated to combine the teachings of Nevarez and Loon because this would have provided a more user friendly manner to communicate with a user (page 43, left col.), in addition to the communications via programming interfaces (APIs) in Nevarez.

As to claim 17, Nevarez teaches invoking (access objects, fig. 2) one of the mapping layer entry-points (via language templates of the UCS). See col. 11, lines 36 – col. 12, line 5. Nevarez does not teach the invocation is via execution of the callback code in response to an entered command.

Loon teaches executing a callback (callback/handlers) in response to an entered command / invocation ('new' operation) (new-handler of C++, signal handler of C). See page 44, section "callback functions". Loon further teaches that executing a callback provides a means to redefine functionalities. See page 46, section "conclusion". Therefore, it would have been obvious to invoke one of the mapping layer entry-points by executing callback code in response to an entered command in Nevarez because mapping from one language to another is a redefinition process in nature.

As to claim 18, Nevarez teaches the entry-points are mapping layer methods (language template libraries of the UCS) that are accessed from the procedural programming language process space though application programming interface ("API") calls (API 218).

As to claim 19, Nevarez teaches the procedural/legacy programming language is C++ (208, 218). Note sections of 2 and 3 of this action for procedural/legacy programming language.

As to claim 20, Nevarez teaches the object-oriented programming language is Java (Java provider 232, Java beans 234, 236) and the object-oriented programming language process space is a Java Virtual Machine (JVM, col. 7, line 65 – col. 8, line 2).

As to claim 1, it is a method claim corresponding to claims 16 and 17, except for the mapping layer executing an algorithm corresponding to the invoked entry-point. Note claims 16-17 for discussion, and Nevarez teaches the mapping layer (UCS) executing an algorithm (object creation) corresponding to the invoked entry-point (language template libraries). See col. 9, line 62 – col. 10, line 4; col. 11, lines 36 – col. 12, line 5.

As to claim 2, Nevarez teaches invoking a class instantiation method (object instantiation function, col. 9, line 62 - col. 10, line 3; col. 11, lines 36-65).

As to claim 3, Nevarez teaches invoking an RMI (Java beans 234, 236, invoked by Java/RMI, col. 7, line 65 – col. 8, line 2).

As to claim 4, note discussion of claim 19.

As to claim 5, Nevarez teaches the object-oriented programming language is Java (Java provider 232, Java beans 234, 236).

As to claim 8, Nevarez teaches the entry-points (language template libraries) are methods of the mapping layer (language template libraries as part of UCS product 224). See discussion of claim 16 with respect to entry-points.

As to claim 9, Nevarez as modified teaches (Loon) returning data to the calling process space (pass client_data to the application, page 45, left col., lines 1-26). When the teachings of Nevarez and Loon are combined, it would have been obvious that the calling process space would have been implemented by the procedural programming language process space. Note discussion of claim 16 for a motivation for combine.

As to claims 10-14, these are the program product claims of claims 1-5, respectively, thus note claims 1-5 for discussions.

7. Claims 6, 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nevarez et a in view of Loon as applied to claims 5 and 14 and further in view of Xia (U S Pat. 6,542,900).

As to claim 6, Nevarez does not teach the mapping layer is accessed through a Java Native Interface ("JNI") and invoking one of the entry-points comprises invoking a JNI API, wherein the JNI API call invokes one of the entry-points.

Xia teaches mapping between a procedural/legacy object-oriented programming language process space (410) and an object-oriented programming language process space (404), wherein a mapping layer (Java wrapper 408) is accessed through a Java Native Interface ("JNI") (JNI 414). See col. 5, line 66 – col. 6, line 6. Xia teaches invoking a JNI API (JNI 414) which invokes one of the entry-points (Java methods corresponding to functions in space 410). See col. 4, lines 57-65. It is noted that by definition, JNI, the invocation API in particular, provides a bi-directional communication interface between a non-Java environment and a Java environment.

Therefore, it would have been obvious to access the mapping layer through JNI to invoke one of the entry-points. One of ordinary skill in the art would have motivated to combine the teachings of Nevarez as modified with Xia because this would have allowed object mapping mechansim, such as Nevarez's, to be distributed and transparent (col. 1, line 60 – col. 2, line 9).

As to claim 15, it is the program product claim of claim 6, thus note claim 6 for discussion.

8. Claim 7 is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

9. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Sue Lao whose telephone number is (703) 305-9657. A voice mail service is also available at this number. The examiner's supervisor, SPE Meng-Ai An, can be reached on (703) 305 9678. The examiner can normally be reached on Monday - Friday, from 9AM to 5PM. The fax phone number for the organization where this application or proceeding is assigned is (703) 872 9306.
Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-9600.

Sue Lao SueLao
June 9, 2004